Exemplo de configuração de serviços de telefone externo seguro

Contents

Introduction

Este documento descreve como configurar o Secure External Phone Service. Essa configuração pode funcionar com qualquer serviço de terceiros, mas para demonstração, este documento usa um servidor remoto do Cisco Unified Communications Manager (CUCM).

Contribuído por Jose Villalobos, engenheiro do Cisco TAC.

Prerequisites

Requirements

A Cisco recomenda que você tenha conhecimento destes tópicos:

- CUCM
- certificados CUCM
- Serviços de telefone

Componentes Utilizados

As informações neste documento são baseadas nestas versões de software e hardware:

- CUCM 10.5.X/CUCM 11.X
- Os telefones Skinny Client Control Protocol (SCCP) e Session Initiation Protocol (SIP) registram-se no CUCM
- O laboratório usa certificados de Nome alternativo do assunto (SAN).
- O diretório externo estará nos certificados SAN.
- Para todos os sistemas neste exemplo, a autoridade de certificação (AC) será a mesma, todos os certificados utilizados são sinal CA.
- O Domain Name Server (DNS) e o Network Time Protocol (NTP) precisam ser configuração de propriedade e estar funcionando.

The information in this document was created from the devices in a specific lab environment. All of

the devices used in this document started with a cleared (default) configuration. Se a sua rede estiver ativa, certifique-se de que você entende o impacto potencial de qualquer alteração.

Produtos Relacionados

Este documento também pode ser usado com as seguintes versões de hardware e software:

- CUCM 9.X/10.X/11.X

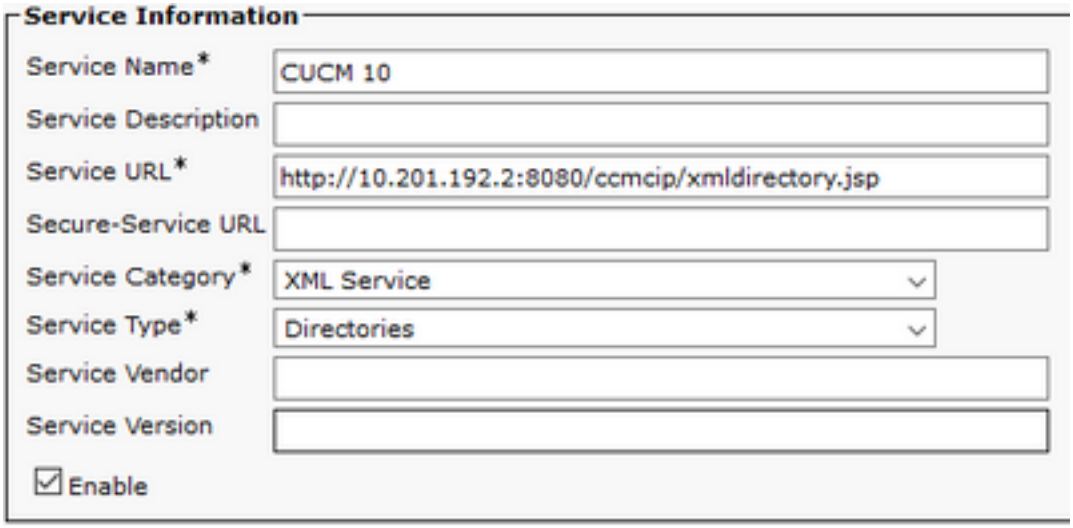
Configuration Steps

Etapa 1. Configure a URL do serviço no sistema.

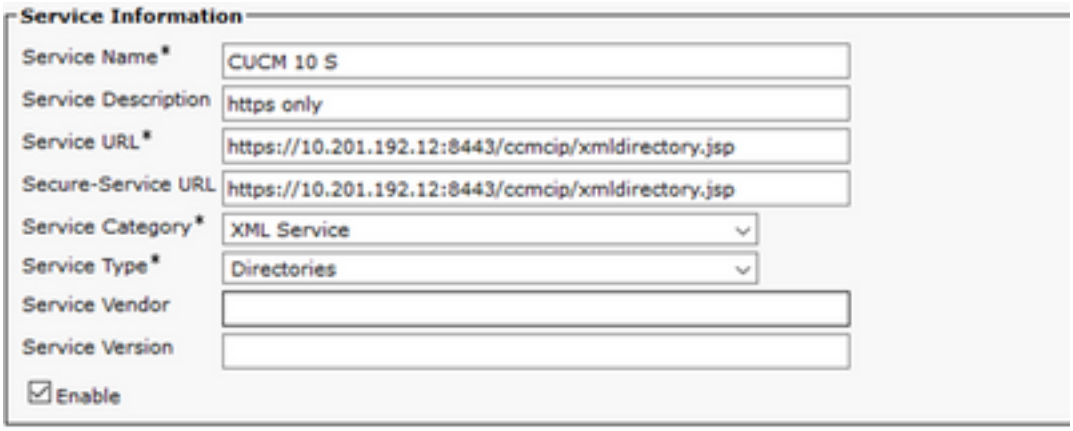
Configure o protocolo HTTP e o protocolo HTTPS como prova de conceitos. A ideia final é usar somente o tráfego HTTP seguro.

Navegue até **Device> Device Settings> Phone service> Add new**

Somente HTTP



Somente HTTPS

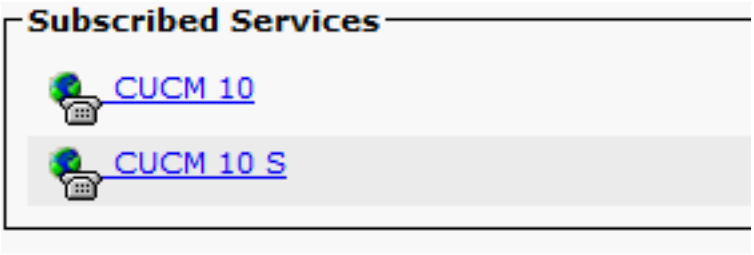


> **aviso:** se você adicionar a verificação de **assinatura do Enterprise,** a etapa dois pode ser ignorada. No entanto, essa alteração redefine todos os telefones, portanto, certifique-se de

que você entendeu o impacto potencial.

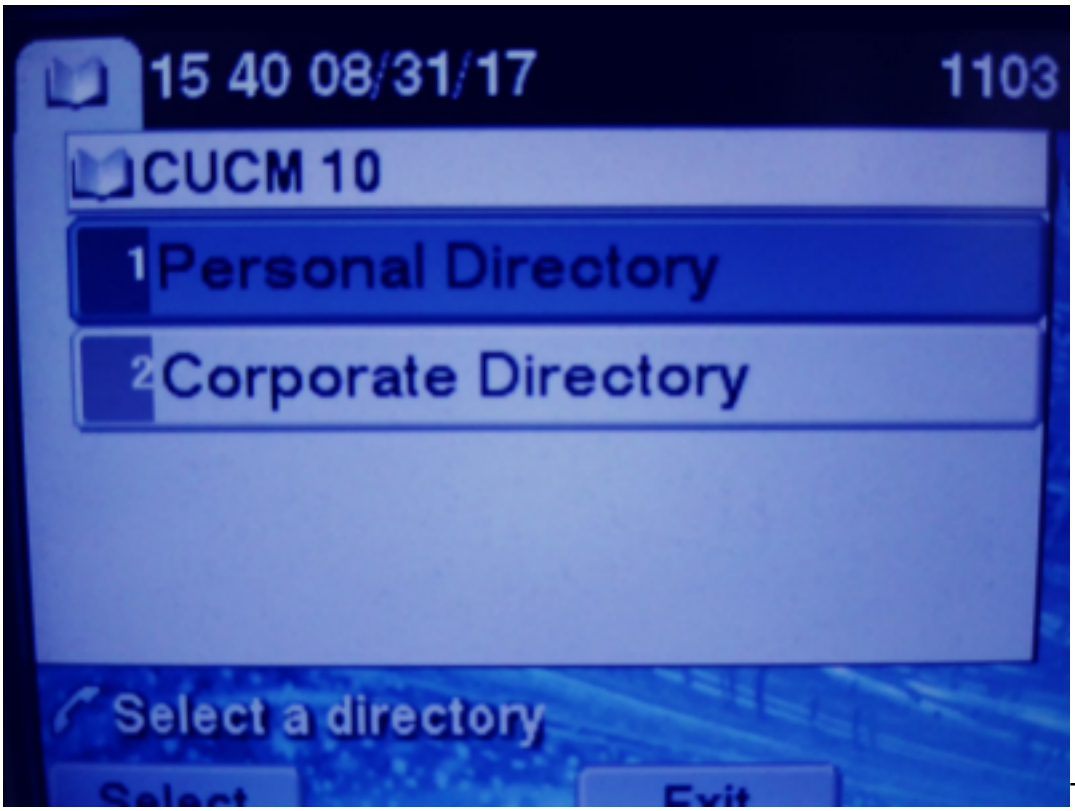
Etapa 2. Inscreva os telefones nos serviços.

Natigate to **Device>Phone>>serviço Subscriber/Unsubscribe.**



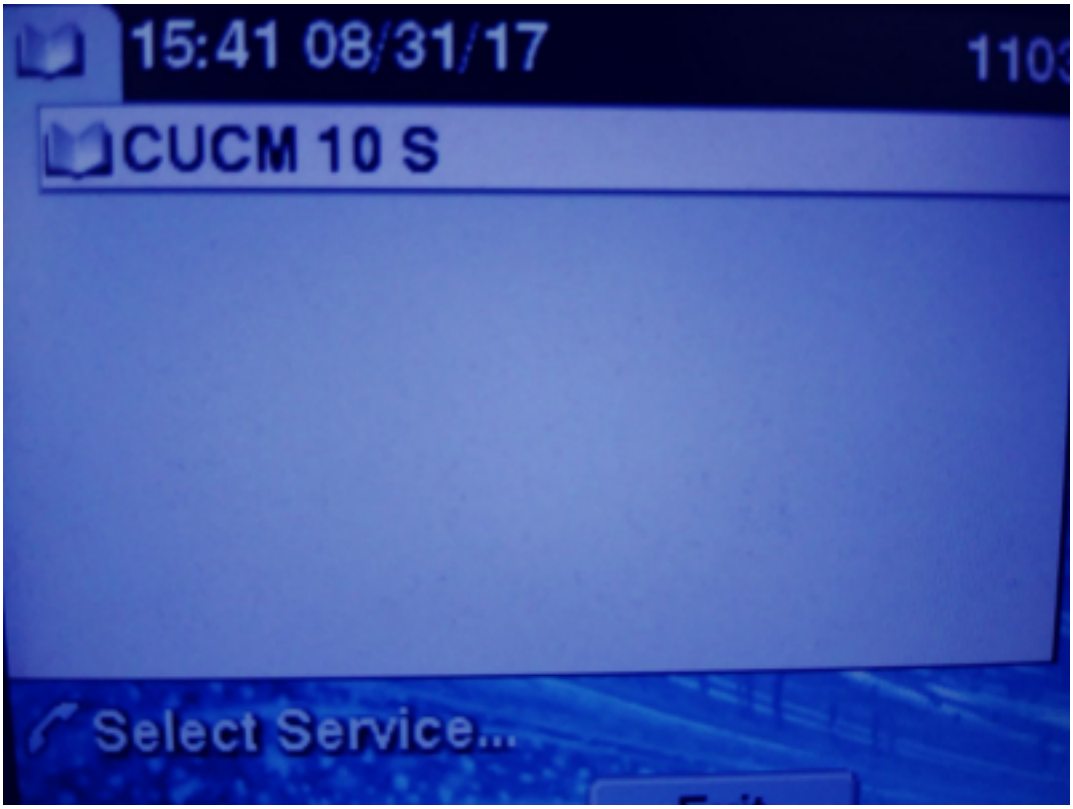
Nesse ponto, se o aplicativo oferecer HTTP, você deve conseguir acessar o serviço, mas https ainda não está ativado.

HTTP



TTP

HTTPS



O HTTPS mostrará um **"Host não encontrado"** devido ao fato de que o serviço TVS não pode autenticar isso para o telefone.

Etapa 3. Carregue os certificados de Serviço Externo no CUCM.

Carregue o Serviço Externo como **Confiança Tomcat apenas**. Verifique se os serviços estão redefinidos em todos os nós.

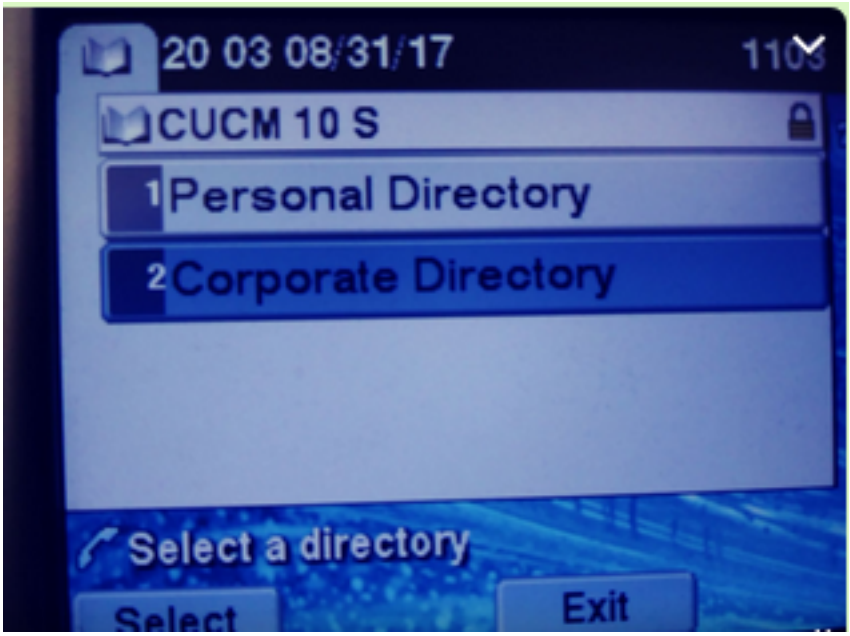
Esse tipo de certificado não é armazenado no telefone, mas o telefone deve verificar com o serviço TVS para ver se ele estabelece a conexão HTTPS.

Navegue até **OS admin> Certificate> Certificate upload.**

A partir do SSH, redefina o serviço CUCM Tomcat em todos os nós.

```
admin:utils service restart Cisco Tomcat
Do not press Ctrl+C while the service is restarting. If the service has not rest
arted properly, execute the same command again.
Service Manager is running
```

Após essas etapas, os telefones devem ser capazes de acessar o serviço HTTPS sem problemas



Perguntas frequentes (FAQ)

Depois que os certificados são trocados, o HTTPS ainda falha com "host não encontrado".

-Verifique o nó onde o telefone está registrado e certifique-se de ver o certificado de terceiros no nó.

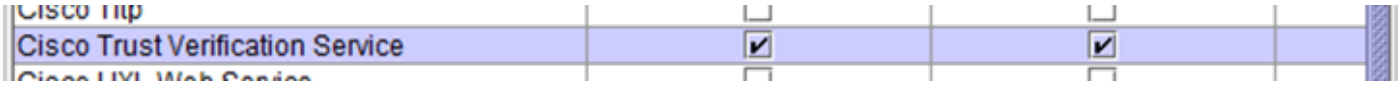
-Redefina o tomcat no nó específico.

-Verifique o DNS, verifique se o nome comum (CN) do certificado pode ser resolvido.

Troubleshooting

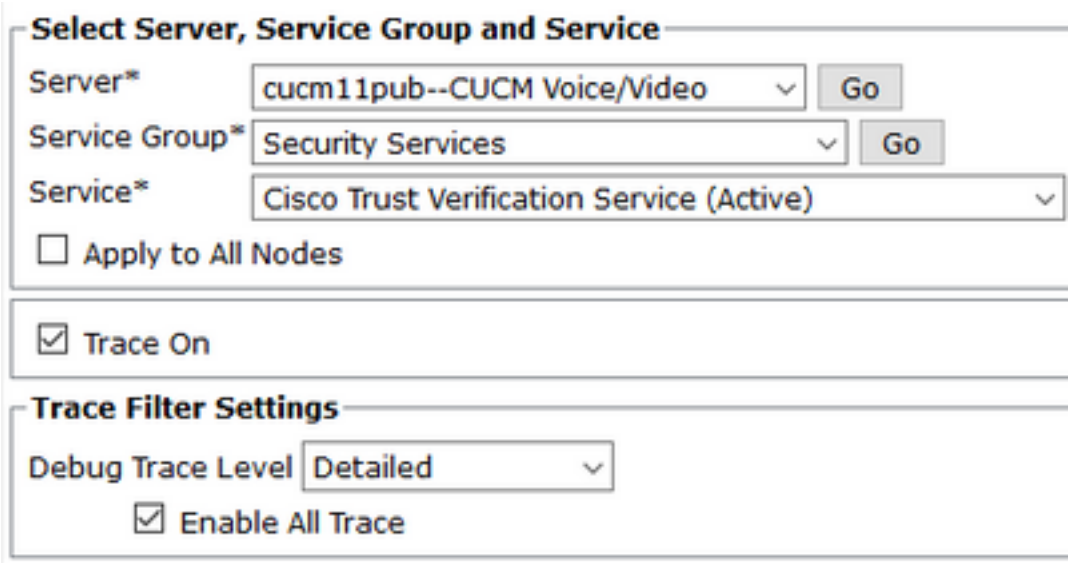
Coletar registros TVS do CUCM deve fornecer boas informações

Navegue até **RTMT>Sistema>Central de rastreamento e registro > Coletar arquivos de log**



> **Note**: Colete registros de todos os nós e certifique-se de que os registros TVS estejam definidos como detalhados.

Logs TVS definidos para detalhado



Exemplo de rastreamento

```
11:17:38.291 |   debug CTVSChangeNotifyServer::ProcessChangeNotification () -
CDBString=<msg><type>DBL</type><table>certificate</table><tableid>46</tableid><action>I</action>
<user>repl</user><time>1504203458</time><new><cdrserver>2</cdrserver><cdrtime>1504203457</cdrtim
e><pkid>e6148ee3-3eb5-e955-fa56-
2baa538a88fb</pkid><servername>cucm11pub</servername><subjectname>CN=10.201.192.12,OU=RCH,O=Cisc
o,L=RCH,ST=Tx,C=US</subjectname><issuername>CN=pablogon-
CA,DC=rcdncollab,DC=com</issuername><serialnumber>3d00000008230ded92f687ec03000000000008</serial
number><certificate></certificate><ipv4address>10.201.192.13</ipv4address><ipv6address></ipv6add
ress><timetolive>NULL</timetolive><tkcertificatedistribution>1</tkcertificatedistribution><ifx_r
eplcheck>6460504654345273346</ifx_replcheck></new></msg>
11:17:38.291 |   debug CTVSChangeNotifyServer::ProcessChangeNotification () - Database table
"certificate" has been changed
11:17:38.291 |   debug CTVSChangeNotifyServer::ProcessChangeNotification () - Looking up the
roles for
11:17:38.291 |   debug Pkid : fead9987-66b5-498f-4e41-c695c54fac98
11:17:38.291 |   debug CTVSChangeNotifyServer::ProcessThreadProc () - Waiting for DBChange
Notification
11:17:38.300 |   debug CTVSChangeNotifyServer::ProcessThreadProc () - DBChange Notification
received
11:17:38.300 |   debug CTVSChangeNotifyServer::ProcessChangeNotification () -
CDBString=<msg><type>DBL</type><table>certificatetrustrolemap</table><tableid>50</tableid><actio
n>I</action><user>repl</user><time>1504203458</time><new><cdrserver>2</cdrserver><cdrtime>150420
3457</cdrtime><pkid>5ae6e1d2-63a2-4590-bf40-1954bfa79a2d</pkid><fkcertificate>e6148ee3-3eb5-
e955-fa56-
2baa538a88fb</fkcertificate><tktrustrole>7</tktrustrole><ifx_replcheck>6460504654345273346</ifx_
replcheck></new></msg>
11:17:38.300 |   debug CTVSChangeNotifyServer::ProcessChangeNotification () - Database table
"certificatetrustrolemap" has been changed
11:17:38.300 |   debug CTVSChangeNotifyServer::ProcessThreadProc () - Waiting for DBChange
Notification
11:17:46.811 |   debug updateLocalDBCache : Refreshing the local DB certificate cache
11:34:00.131 |   debug Return value after polling is 1
11:34:00.131 |   debug FD_ISSET i=0, SockServ=14

11:34:00.131 |   debug Accepted TCP connection from socket 0x00000014
```